**Plantilla Diseño del Sistema**

1. **Introducción**
   1. **Propósito del sistema.**

Women store aplicativo para la venta y distribución de productos de maquillaje y cuidado personal por medio de una app o pagina web con la ventaja de llevar estos productos a domicilio o si el cliente decide recogerlo en una tienda física también los negocios podrán solicitar productos al por mayor y con un buen precio.

**1.2. Objetivos del diseño**

*1 llegar al 100% de la población de Soacha*

*2.ser conocidos por el país*

*3.ofrecer el mejor servicio y atención al cliente*

**1.3. Definiciones, acrónimos y abreviaturas**

*Este apartado proporciona las definiciones de todos los términos, acrónimos y abreviaturas utilizadas a lo largo del documento y que permiten una interpretación correcta del mismo. Se han de incluir los términos técnicos, caso de uso por ejemplo, y los específicos del entorno del sistema, lector de bandas por ejemplo. Es conveniente ordenarlos alfabéticamente.*

**1.4. Referencias**

*Referencias a otros documentos e información de trazabilidad – documento/s de análisis de requisitos relacionados, sistemas existentes, restricciones que afectan a la arquitectura software, etc.*

**2. Representación de la arquitectura.**

**2.1 Metas y restricciones de la Arquitectura**

*Describir los requisitos y objetivos del software que tienen un impacto significativo sobre la arquitectura; por ejemplo, seguridad, privacidad, portabilidad, distribución, rendimiento, escalabilidad, reutilización.*

*Como metas de este sistema de información se puede esbozar lo siguiente:*

*-Sistema de información (Instituciones Educativas ABC, Sistema de Servicios al Hogar XYZ, Escuela de Futbol ABC, … según su proyecto), destinado a prestar servicios a las empresas o instituciones siguientes: Colegio ABC, Empresa XYZ, Empresa de organización de Eventos XXX, con soporte y servicio las 24 horas del día, con autonomía y servicios permanentes, ceñidos a los prototipos de seguridad y calidad internacionales según documento IEEE1016. Actualmente esta solución va a complementar los servicios de este otro sistema (nombrarlo si lo hay o ignorar esto si no lo hay).*

*En el esquema de despliegue técnico tendrá:*

* Interfaz para ser usada en ambiente web con conexión a internet.
* Uso de dominio de internet
* Lenguajes y tecnologías en uso: HTML, JAVA, PHP, CSS, JavaScript, Ajax y los frameworks
* Los servidores deben ser capaces de atender consultas concurrentemente.
* El sistema se diseñará según un modelo cliente/servidor.
* El sistema deberá tener un diseño e implementación sencilla, independiente de la plataforma o del lenguaje de programación.

**2.2 Reutilización**

*Si se utiliza programación orientada a objetos (OO), se debe proporcionar una ligera visión de cómo puede beneficiar al proyecto la incorporación de componentes reutilizables, frameworks y patrones de diseño.*

*La utilización del modelo orientado a objetos permitirá la reutilización de los conceptos de superclases para (nombrar las clases (superclases) que tienen herencia en otras) haciendo más productiva la codificación, empleando atributos que se podrán llamar para ser usados en otros apartes de la solución ahora o en futuras implementaciones*

1. **Vista lógica**

*Aquí se describe la descomposición funcional de la aplicación basándose en una ordenación lógica de los requisitos de la aplicación. Los aspectos de la aplicación con una funcionalidad similar se deben agrupar en un subsistema. Se deben representar las dependencias entre los subsistemas.*

*Nombre aquí todos los requisitos funcionales (del informe IEEE830 – Especificación de Requisitos del trimestre1) que tiene la aplicación, agregue el diagrama de clases*

*Agregue el diagrama de paquetes de la solución*

*Agregue el diagrama de actividades para:*

*-Registro de: clientes, usuarios, productos, eventos, entrenadores, administradores de conjunto residencial, residentes del conjunto, evaluaciones o notas del colegio.*

*-Haga un diagrama de actividades de cómo se actualiza un: usuarios, productos, eventos, entrenadores, administradores de conjunto residencial, residentes del conjunto, evaluaciones o notas del colegio. Estos son similares a los de registro, solo que la actualización hace una consulta del dato antes de proceder a actualizarlo, si encuentra (si existe) un registro procede a actualizarlo, de lo contrario en una decisión arroja el mensaje “Registro no existe”*

* 1. **Identificación de Subsistemas**

*Descomponer el sistema en subsistemas funcionales, para realizar esto se deben agrupar las tareas del sistema que presenten propiedades en común y que presten un mismo tipo de servicio o funcionalidad.*

* *Se debe procurar realizar un número pequeño de subsistemas*
* *Cada subsistema debe tener un mecanismo de colaboración con el resto.*
* *Utilizar los diagramas de paquetes para su representación*
* *Al interior de cada paquete se puede incorporar las clases que lo componen*
  1. **Estructuración por capas.**

*Una vez que se han determinado los subsistemas, agruparlos mediante una estructuración por capas, donde cada una determina un nivel de abstracción.*

* *Determinar el número de capas a implementar, acorde con el sistema*
* *Nombrar cada capa y definir su funcionalidad*
* *Asignar los subsistemas / paquetes /clases que corresponden con cada capa*
* *Se pueden utilizar diagramas de paquetes*

**Diagrama de Clases del diseño**

*Presentar las clases significativas de la arquitectura y describir sus responsabilidades, así como las relaciones importantes basadas en la arquitectura, operaciones y atributos. Especificar los patrones de diseño que pueden estar incluidos.*

1. **Vista del proceso**

*Describe la descomposición del sistema en procesos, se debe representar la información solicitada utilizando diagramas de secuencia para tres de los casos de uso más representativos del negocio.*

* 1. **Diagramas de secuencia**

*Se debe representar la información solicitada utilizando diagramas de secuencia específicos del proyecto (diagramas de interacción de objetos), preferiblemente utilizando la notación UML. Donde sea posible, los diagramas explican el proceso de interacción requerido por los casos de uso principales.*

1. **Vista de datos**

*Describe el modelo de datos del sistema que se va a desarrollar. Se realiza la identificación a través de diagramas relacionales que presenten el modelo de base de datos a implementar y su descripción usando el diccionario de datos.*

* 1. **Modelo de datos**

*Se presenta el modelo relacional de la base de datos a través de un diagrama, donde se identifican las tablas, campos y relaciones entre tablas que fueron definidas para almacenar los datos del sistema de información.*

* 1. **Diccionario de datos**

*Se puede construir el formato específico que detalle para cada una de las tablas los campos, tipos de datos y restricciones o elementos de integridad a ser incorporados. También se pueden utilizar los generados por herramientas CASE.*

1. **Vista de Interacción**

*Se deben presentar las distintas interacciones con las que contará la aplicación, para lo cual se presenta las “interfaces de usuario”.*

* 1. **Interfaces de Usuario.**

*Lista y describe las interfaces de usuario de la aplicación, especificarlas por caso de uso.*

* 1. **. Mapa de Navegación**

*Presenta la estructura global de navegación de la aplicación. (Escriba en su informe este texto para no obligar a repetir la información: “Revise el numeral 6.”)*

1. **Vista de seguridad**

*Describir los distintos elementos y sistemas de seguridad con los que cuenta el software.*

***(Explicar que la aplicación necesitará de un sistema de copias de seguridad diaria, semanal, mensual, anual para conservar la seguridad de la información)***

***(Explicar que se necesita de un sistema de seguridad de datos automatizado para los inicios de sesión o para envíos de datos, ver este reporte:***

[***https://www.redeszone.net/2019/03/22/hash-vs-cifrado-contrasenas/***](https://www.redeszone.net/2019/03/22/hash-vs-cifrado-contrasenas/)***)***

***(Ver este reporte para razonar acerca de la seguridad en cifrados:***

[***https://www.redeszone.net/2010/11/16/criptografia-algoritmos-de-cifrado-de-clave-asimetrica/***](https://www.redeszone.net/2010/11/16/criptografia-algoritmos-de-cifrado-de-clave-asimetrica/)***)***

***Sistema de Acceso***

*Se debe definir de forma clara el acceso al sistema: nivel de seguridad de acceso, empleo de las claves de acceso. Incluir la segmentación de procesos, perfiles y roles y los mecanismos de autenticación a implementar en el sistema*

***Cifrado de datos***

*Existe información en la base de datos que debe ser cifrada o encriptada. Si es así qué algoritmos de cifrado se utilizarían o recomiendan según estos link de consulta sugeridos*

***(Ver este artículo de seguridad:***

[***https://www.redeszone.net/2010/11/09/criptografia-algoritmos-de-autenticacion-hash/***](https://www.redeszone.net/2010/11/09/criptografia-algoritmos-de-autenticacion-hash/)***), el sistema de cifrado SHA-2 con SHA-512 podría ser útil para su aplicación, no ser recomendaría MD5 ni SHA1 porque ha encontrado casos de “colisión de datos”.***

1. **Vista de Implementación**

*Describe la estructura general del modelo de implementación y la descomposición del sistema. Aquí viene el diagrama de despliegue sin incluir el software de cada nodo.*

* 1. **Herramientas de Desarrollo e implementación**

*Describe las herramientas tecnológicas que se deben utilizar para el desarrollo del sistema, incluye: IDE, Lenguaje de Programación, base de datos, framework, etc. Además, el software que se requiere para su ejecución y puesta en marcha: servidor de aplicaciones/web, Sistema de base de datos, Librerias, plugins, etc.*

*Explicar que se necesita usar el software Netbeans que es un IDE (Integrated Development Environment o Entorno de Desarrollo Integrado), con Netbeans usará JAVA Enterprise Edition (EE).*

*También necesita usar el paquete XAMPP que contiene la base de datos MySQL una de las herramientas más versátiles para aplicaciones web, el lenguaje de programación PHP para hacer aplicaciones web, TomCat para repositorio de herramientas Java, XAMPP contiene el “servidor web virtual” Apache con el cual desarrolla y emula sitios web desde un solo equipo.*

*Puede usar “Marcos de Trabajo” o frameworks y otras herramientas como:*

*-Laravel (Marco de trabajo para PHP)*

*CodeIngniter (Marco de trabajo para PHP)*

*Spring MVC (Marco de trabajo para Java)*

*JSF (Marco de trabajo para Java)*

*GWT (Marco de trabajo para Java)*

*Spring Boot (Marco de trabajo para Java)*

*Marcos de trabajo diversos para JavaScript: Angular JS, React JS, Node JS, Vue JS.*

*Bibliotecas diversas para JavaScript como: Chart JS, Create JS, Moment JS.*

*Herramientas para sitios web responsive como Bootstrap, Foundation.*

* 1. **Paquetes/Componentes**

*Describir el modo principal de comunicación entre los procesos del sistema operativo. Incluir diagramas de componentes. Hacer el diagrama de despliegue incluyendo los paquetes de software que hemos sugerido en cada capa*

* 1. **Despliegue**

*Describir la configuración de la plataforma física (procesador/almacenamiento) en la que el software va a ser desplegado. Si el sistema se va a desplegar en varios sitios, proporcionar una vista de despliegue para cada sitio diferente. Como mínimo, para cada configuración, se deben indicar los nodos físicos (ej.: ordenadores, CPUs, memorias) que ejecutan el software y sus interconexiones (ej.: bus, topología LAN, punto a punto, WAN). (Hacer diagrama de despliegue con el software que vendrá en cada nodo)*

*Los elementos de hardware de la solución mínimos son:*

* Procesador Intel o AMD 2.0 GigaHertz 4 núcleos o superior, recomendable Intel Core i7 o Ryzen 7
* Memoria mínima de 4 GigaBytes.
* Adaptadores de red Ethernet (con cableado UTP)
* Mouse óptico USB o inalámbrico.
* Teclado PS/2 o USB.
* Adaptador o tarjeta de vídeo DVI + DisplayPort + VGA (opcional) o chip de vídeo integrado con memoria expandible a 1 GB
* Monitor LCD 14 pulgadas widescreen
* Unidad de almacenamiento 1000 GB en formado HDD (Disco duro estándar mecánico) o recomendable SSD (dispositivo sólido de almacenamiento)

***Resumen de Costos asociados***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Item*** | ***Cantidad*** | ***Descripción*** | ***Valor Unitario*** | ***Total*** |
|  | *3* | *Computador laptop o desktop con condiciones técnicas de hardware según numeral 8.3.* | *3.200.000* | *9.600.000* |
|  | *3* | *Suite de Oficina (Procesador de palabra, hoja de cálculo, presentaciones)* | *350.000* | *1.050.000* |
|  | *1* | *Contrato de hosting básico* | *500.000\** | *500.000* |
|  | *1* | *Servicio soporte técnico de software y hardware* | *4.000.000\** | *4.000.000\** |

*\* Servicio por 12 meses.*

*Incluir un mapeo entre los procesos de la vista de proceso y los nodos físicos. La notación preferida es UML para la vista de despliegue.*

* 1. **Vista de Administración**

*Describe las distintas opciones de la administración del software.*

Hacer una vista de las interfaces de usuario del sistema. (Aportar las interfaces java o css para apoyar el servicio del sistema.

-Hacer un modelo de menú java o css con una interfaz para hacer copia de seguridad de la base de datos completa.

-Hacer un modelo de menú java con una interfaz para hacer Copia de seguridad (Backup) del sistema. Ejemplo, un menú que se titule: Copias de seguridad, con tres opciones:

1. Copia completa,
2. Copia de la base de datos.
3. Copia de los módulos de software

-Hacer un modelo de menú java con una interfaz para hacer Restauración de una copia de seguridad (Restore) del sistema. Ejemplo, un menú que se titule: Restaurar Sistema, con tres opciones:

1. Restauración completa,
2. Restauración de la base de datos.
3. Restauración de los módulos de software
4. Hacer mantenimiento de tablas (eliminar registros temporales del sistema)

Utilicemos estos documentos de ayuda:

1. Hacer Maquetas con Balsamiq FireFrames:

<https://www.youtube.com/watch?v=4WqfxF1QjTM>

<https://www.youtube.com/watch?v=6vXjEd6BWgo>

<https://www.youtube.com/watch?v=Y398S0IhCzE>